บทความพิเศษ

พรินเตอร์ไดรเวอร์

ทางออกของการใช้เครื่องพิมพ์ต่างยี่ห้อและรหัสไม่ตรงกัน

SOFTMAIL

"ทำความเข้าใจกับโปรแกรมที่เรียกว่าพรินเตอร์ไดรเวอร์ ซึ่งทำหน้าที่ช่วยให้พรินเตอร์ยี่ห้อต่าง ๆ ที่มีรหัสในการควบคุมการสั่งพิมพ์ไม่เหมือนกันสามารถพิมพ์งานกับโปรแกรมสำเร็จรูปต่าง ๆ ได้สำเร็จ"

บุญภพ ลิมมโนกุล

ในโลกของไมโครคอมพิวเตอร์เราจะพบว่าพวกโปรแกรมต่าง ๆ ที่ใช้งานอยู่เมื่อ ๕ ปีที่แล้ว ยังสามารถนำมาใช้กับเครื่องคอมพิวเตอร์ปัจจุบันได้ ตั้งแต่โปรแกรมที่วิ่งอยู่บนเครื่องพีซี พีซีเอกซ์ทีและรุ่นเทอร์โบ ก็ยังสามารถนำมาวิ่งบนเครื่องไมโครคอมพิวเตอร์รุ่นใหม่ ๆ ได้เช่น พีซีเอที เนื่องจากในระดับภาษาเครื่อง ซีพียูนั้นสามารถรับรู้คำสั่งของซีพียูรุ่นเก่าได้ เราอาจจะพูดได้ว่า เราสามารถนำคำสั่งของซีพียูมาใช้กันได้ระหว่างคำสั่งของซีพียูรุ่นเก่ากับซีพียูรุ่นใหม่ที่อยู่ในตระกูล 80xxx ของ

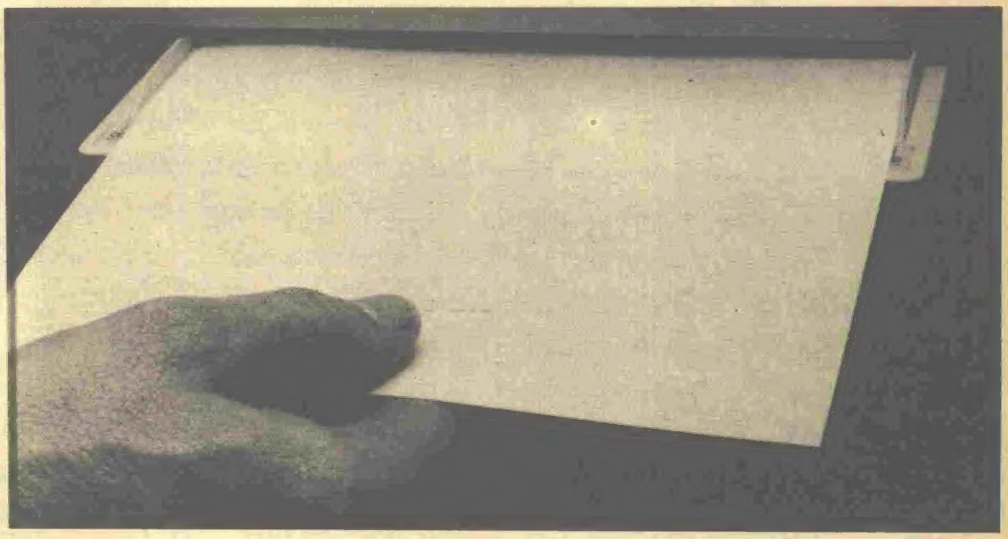
อินเทล แต่อาจจะนำโปรแกรมสมัยใหม่ที่เขียนขึ้นมาเพื่อใช้กับเครื่องคอมพิวเตอร์สมัยใหม่ไปใช้กับเครื่องพีซีที่เราซื้อเมื่อ 5 ปีที่แล้วไม่ได้ เพราะบางคำสั่งในระดับภาษาเครื่องแล้วซีพียูรุ่นเก่าไม่รู้จัก ยกตัวอย่างเช่น คำสั่ง PUSHA เป็นคำสั่งให้นำค่าต่าง ๆ ในรีจิสเตอร์ของซีพียู 80286 เก็บไว้ในสแต็ก แต่สำหรับซีพียู 8088 หรือ 8086 เราจะต้องเขียนคำสั่ง PUSH รีจิสเตอร์อยู่หลายบรรทัด จึงจะสามารถทำงานเท่ากับคำสั่ง PUSHA ของซีพียู 80286 เพียงคำสั่งเดียวเป็นต้น แต่ถ้าเราจะมองในเรื่องของการพิมพ์รายงานทางเครื่องพิมพ์ ตลาดในเมืองไทยสมัยก่อนเรามีเครื่องพิมพ์แบบเข็ม แบบจานหมุน แบบหัวฆ้อน แล้วพัฒนามาเรื่อยจนถึงเครื่องพิมพ์แบบเลเซอร์แล้วก็ยังพัฒนาต่อไปอีก แต่สิ่งหนึ่งที่เราจะพบนั้นคือ คำสั่งบางคำสั่งที่ใช้กับเครื่องพิมพ์แบบเข็มหรือดอตแมตริกซ์ ไม่สามารถที่จะนำมาใช้กับเครื่องพิมพ์เลเซอร์ได้ ยกตัวอย่างเช่น คำสั่งการขีดเส้นใต้ของเครื่องพิมพ์แบบเข็มเป็นอีกแบบหนึ่ง หมายความว่าหากที่ทำงานของคุณได้เครื่องพิมพ์แบบเลเซอร์มา คุณก็จะต้องเสียเวลาส่วนหนึ่งที่จะพัฒนาโปรแกรมที่เกี่ยวกับการพิมพ์รายงานให้สามารถพิมพ์รายงานบนเครื่องพิมพ์เลเซอร์ได้ เมื่อคุณพัฒนาเสร็จแล้วคุณจะพบว่าโปรแกรมของคุณมีเวอร์ชันเพิ่มขึ้นเป็นเวอร์ชันใหม่ ก็จะเพิ่มความยุ่งยากและเป็นภาระในการดูแลการทำงานของเครื่องคอมพิวเตอร์แล้วยังต้องคอยดูแลโปรแกรมคนละเวอร์ชันอีก หากว่าในระบบงานของคุณมีโปรแกรมเป็นร้อยโปรแกรม การทำโปรแกรมที่เกี่ยวกับการพิมพ์เข้าไปทดแทนก็จะเพิ่มภาระมากขึ้นเป็นทวีคูณ เกิดวันใดเครื่องพิมพ์รุ่นใหม่เสียและไม่สามารถที่จะซ่อมกลับคืนมาใช้ดังเดิมได้ภายในหนึ่งวัน คุณก็ต้องนำเอาเครื่องพิมพ์รุ่นเก่ามาทดแทนเพื่อให้สามารถพิมพ์รายงานได้ แล้วโปรแกรมที่เกี่ยวกับการพิมพ์รายงานก็ต้องนำโปรแกรมเก่ามาใช้ ซึ่งจะทำให้เกิดข้อผิดพลาดได้ กว่าคุณจะแก้ปัญหาได้ลงตัวคุณก็จะสิ้นเปลืองเวลาไปพอพอสมควร

เรามาดูว่าในเรื่องของการพิมพ์รายงานนั้นไม่ว่าจะอยู่ในรูปของตารางหรือในรูปของการพิมพ์จดหมาย ก็พอที่จะจัดกลุ่มของความต้องการพิมพ์ได้ 5 กลุ่มคือ

1. การปรับหมุนหรือการเลื่อนกระดาษขึ้นใน 1 บรรทัด รวมทั้งการขึ้นหน้าใหม่
2. รูปแบบตัวอักษรแบบหยาบหรือแบบละเอียด (Draft & NLQ)
3. โหมดของการพิมพ์ เช่น ตัวอักษรแบบตัวหนา ตัวจิ๋ว การขีดเส้นใต้ และตัวเอียง
4. การเลือกชุดตัวอักษร หรือการดาวน์โหลดฟอนต์ของตัวอักษร
5. การแม็พ (MAP) ฟอนต์ของตัวอักษรที่นำมาพิมพ์เส้นตาราง และรหัสของฟอนต์ที่ต่างกัน เรามาดูในรายละเอียดปลีกย่อยของแต่ละกลุ่มเพิ่มเติมว่าต่างกันอย่างไร

การปรับหมุนหรือเลื่อนกระดาษ

การปรับหมุนหรือเลื่อนกระดาษ เราจะพบว่าความละเอียดของการเลื่อนกระดาษในแต่ละบรรทัดขึ้นกับความสามารถเครื่องพิมพ์ของบริษัทผู้ผลิตที่ออกแบบแต่ละรุ่น ยกตัวอย่างเช่น คำสั่งเซตระยะห่างระหว่างบรรทัด เมื่อเครื่องพิมพ์ทำการพิมพ์ข้อความจบไป 1 บรรทัด จะต้องปรับหมุนกระดาษขึ้นเมื่อได้รับคำสั่งไลน์ฟีด (LF) และรีเทอร์น (CR) เราจะพบว่าตามคู่มือที่มากับเครื่องพิมพ์จะบอกถึงความสามารถในการหมุนกระดาษได้ละเอียดถึงกี่ส่วนใน 1 นิ้ว ยกตัวอย่างเช่น คำสั่งปรับระยะห่างระหว่างบรรทัด เช่น ESC 3 n (n/216 inch line spacing) เป็นคำสั่งที่ส่งให้เครื่องพิมพ์รู้ว่าการเลื่อนกระดาษขึ้นในแต่ละบรรทัด หลังจากเจอคำสั่งไลน์ฟีดเป็นระยะห่างกันเท่ากับ n/216 สำหรับเครื่องพิมพ์ของ EPSON รุ่น LX 800 และรุ่น FX 100 แต่ถ้าเป็นรุ่น LQ 1500, รุ่น LQ 550 คำสั่งเดียวกันคือ ESC 3 n (n/180 inch line spacing) เราจะพบว่าเมื่อใช้คำสั่งนี้กับเครื่องพิมพ์คนละรุ่นถึงแม้ว่าจะเป็นบริษัทเดียวกัน การเลื่อนกระดาษในแต่ละบรรทัดของเครื่องพิมพ์ด้วยโปรแกรมพิมพ์รายงานตัวเดียวกัน โดยเฉพาะภาษาไทยแล้วจะมองเห็นข้อแตกต่างนี้ เช่น ในโปรแกรมเรากำหนดค่า n = 20 ดังนั้นในเครื่องพิมพ์รุ่น LX 800 จะเลื่อนกระดาษครั้งละ 20/126 แต่เครื่องพิมพ์รุ่น LQ 500 จะเลื่อนกระดาษไปครั้ง



ละ 20/180 หน่วย เราจะเห็นได้ชัดของความแตกต่างในการพิมพ์รายงาน

รูปแบบตัวอักษรแบบหยาบหรือแบบละเอียด (Draft & NLQ)

รูปแบบตัวอักษรแบบหยาบหรือแบบละเอียด (Draft & NLQ) จะขึ้นกับความสามารถของเครื่องพิมพ์ ถ้าเป็นเครื่องพิมพ์รุ่นแรก ๆ จะไม่สามารถพิมพ์ในรูปของ NLQ ได้ แต่ในปัจจุบันเป็นเรื่องธรรมดาที่เครื่องพิมพ์รุ่นใหม่ ๆ จะต้องทำได้ แต่ทว่าคำสั่งที่จะส่งให้เครื่องพิมพ์บางยี่ห้อจะต่างกัน จะขอยกตัวอย่างเช่น เครื่องพิมพ์รุ่น LQ 550 ของ EPSON จะใช้ ESC x 1 สำหรับการพิมพ์แบบ NLQ โหมด และ ESC X 0 สำหรับการพิมพ์แบบหยาบ (Draft) แต่เครื่องพิมพ์ของไอบีเอ็มรุ่น X 24 และ XL 24 จะใช้ ESC 1 n โดยที่ตัว n นั้นจะมีอยู่หลายตัว เช่น n=0 จะเป็นการพิมพ์แบบ Draft 10 ตัวอักษรต่อนิ้ว แต่ถ้า n=2 จึงจะเป็น LQ ขนาด 10 ตัวอักษรต่อนิ้ว (เครื่องพิมพ์รุ่นนี้ไม่เรียกการพิมพ์โหมดนี้ว่า NLQ แต่จะเรียกว่า LQ เฉย ๆ) เราจะพบว่าความต้องการพิมพ์รูปแบบตัวอักษรจะเริ่มมีปัญหาเพิ่มขึ้นเมื่อเครื่องพิมพ์ที่เราซื้อมาเป็นคนละยี่ห้อ

โหมดของการพิมพ์

โหมดของการพิมพ์ เช่น ตัวอักษรแบบหนา ตัวจิ๋ว การพิมพ์ขีดเส้นใต้ และตัวเอียง การพิมพ์ในโหมดนี้ความจริงยังมีมากกว่านี้แต่จะไม่ขอกล่าวถึงทั้งหมด เครื่องสมัยใหม่มักจะมีความสามารถใกล้เคียงกันในราคาที่ใกล้เคียงกัน เท่าที่พบเช่นเครื่องพิมพ์ของ NEC รุ่น P5260 หรือ P5360 จะใช้คำสั่ง ESC 4 เป็นคำสั่งให้เครื่องพิมพ์พิมพ์ตัวอักษรแบบตัวเอียง และยกเลิกการพิมพ์ตัวเอียงด้วยคำสั่ง ESC 5 แต่เครื่องพิมพ์ของ IBM Proprinter X 24 และ Proprinter XL 24 จะแปลความหมายของคำสั่งดังกล่าวว่า ESC 4 เป็นการกำหนดขอบที่จะเริ่มพิมพ์ตอนบนของกระดาษ (Set Top of form) ส่วน ESC 5n (N=0 cancels, n=1 begins) เป็นการบอกให้เครื่องพิมพ์จัดการในเรื่องของการเลื่อนกระดาษอัตโนมัติ (Automatic Line Feed) ดังนั้นในการพิมพ์รายงานของคุณหากว่าคุณมีความประสงค์จะพิมพ์ตัวเอียง เมื่อคุณมาใช้กับเครื่องพิมพ์อีกบริษัทหนึ่งจะกลายเป็นการทำงานอีกแบบหนึ่ง คุณคงจะไม่ชอบใจแน่

การเลือกชุดตัวอักษร

การเลือกชุดตัวอักษรหรือการดาวน์โหลดฟอนต์ของตัวอักษร เป็นปัญหาอีกอย่างหนึ่งที่คุณไม่สามารถหลีกเลี่ยงได้ การพิมพ์ในโหมดนี้ส่วนใหญ่จะใช้ฟอนต์ของตัวอักษรที่ไม่มีในเครื่องพิมพ์หรืออาจจะเป็นสัญลักษณ์ทางคณิตศาสตร์ เป็นต้น โดยที่คุณจะต้องออกแบบตัวอักษรเอง จะพบข้อแตกต่างกันระหว่างความละเอียดของหัวพิมพ์ชนิดหัวเข็มแบบ 8 เข็ม และ 24 เข็มจะมีข้อมูลของฟอนต์ที่ต่างกัน แต่เราจะพบว่าเครื่องพิมพ์ในระดับ 24 เข็มเองข้อมูลที่ใช้ในการสร้างฟอนต์ยังมีความแตกต่างกัน เช่น เครื่องพิมพ์ของ NEC รุ่น P6/P7 การออกแบบตัวอักษรหรือฟอนต์ในโหมดตัวอักษรแบบละเอียดที่ 10 ตัวอักษรต่อนิ้วในคู่มือจะบอกว่าคุณสามารถใช้จุดตามแนวนอนได้ถึง 32 เข็มโดยที่มีข้อห้ามคือ ระหว่างเข็มในแนวนอนให้เว้น 1 ช่องห้ามชิดติดกัน และกำหนดว่า B = 32 และ A+B+C = 36 แต่สำหรับเครื่องพิมพ์ LQ 550 ของ EPSON การพิมพ์โหมดเดียวกันตามคู่มือจะกำหนดให้ตัวเลขมาเลือก d1 = 23 แต่ว่า d0+d1+d2 มีค่าสูงสุดไม่เกิน 36 ค่า d0 และ d2 นั้นจะเป็นช่องไฟข้างซ้ายและขวาของตัวอักษร เราจะเห็นว่าค่าตัวกลางของเครื่องพิมพ์ NEC รุ่นนี้กำหนดให้ตัว B=32 ในขณะที่เครื่องพิมพ์ของ EPSON กำหนด d1=23 สำหรับคุณที่เคยออกแบบฟอนต์มาแล้วคงจะทราบดีถึงความยุ่งยากการออกแบบตัวอักษร

การแม็พ (MAP) ฟอนต์ของตัวอักษร

การแม็พ (MAP) ฟอนต์ของตัวอักษรที่นำมาพิมพ์เส้นตาราง รหัสของฟอนต์ที่ต่างกันในเรื่องของฟอนต์ตัวตารางมักจะมีปัญหา ตัวอย่างเช่น เครื่องพิมพ์รุ่นแรกที่ใช้รหัสของเกษตรจะใช้ฟอนต์การทำตัวตารางอยู่ที่รหัสแอสกีแถว ๆ 149 และ 150 เครื่องพิมพ์รุ่นใหม่ ๆ ในปัจจุบันแม้ว่าจะเป็นรหัสเกษตรก็ตาม จะถูกย้ายตำแหน่งมาที่แอสกี 128 และ 129 ดังนั้นโปรแกรมของคุณที่กำหนดการตีเส้นตารางจะใช้กับเครื่องรุ่นเก่ารหัสเกษตรก็ควรจะคำนึงถึงเครื่องพิมพ์รุ่นปัจจุบันด้วย อาจจะพิมพ์ไม่ได้หากว่าข้อตกลงในการซื้อขายเครื่องพิมพ์ไม่รัดกุมพอ ปัญหาที่ผมเจอเองคือรหัสภาษาไทยไทยของสมอ.เอง ตัวสระ ไ, ใ, โ ก็ยังมีรุ่นต่อหัวกับไม่ต่อหัว ความจริงเราต้องการพิมพ์คำว่า ไม่ อาจจะได้เพียงแค่ " หรือส่วนหัวของ "ไ" หรือได้ไม่ครบนั้นเอง

จากที่ยกข้อแตกต่างพร้อมตัวอย่างของคำสั่งบนเครื่องพิมพ์มาทั้ง 5 ข้อ ความจริงแล้วยังมีรายละเอียดปลีกย่อยอีกมาก อย่างไรก็ตามนับว่าเป็นความฉลาดของผู้ออกแบบโปรแกรมสำเร็จรูปที่จะเปิดช่องนี้ไว้ให้โดยที่ก่อนจะใช้งานหรือการรันโปรแกรมครั้งแรก โปรแกรมสำเร็จรูปที่ฉลาดจะมีการเก็บสถานะ (flag) ของโปรแกรมเอาไว้ว่าเป็นการรัน

ครั้งแรกหรือไม่ โดยที่งานหยาบวิ่งเข้าหาโปรแกรมเตรียมตัวให้พร้อม (Set up) ให้เราเลือกว่าเราจะใช้เครื่องพิมพ์ของบริษัทอะไร รุ่นไหน เช่น คุณมีเครื่องพิมพ์ รุ่น LX 800 ของ EPSON ผลิตออกมาวางขายในตลาดภายหลังพวกโปรแกรมสำเร็จรูปได้วางขายในตลาดเรียบร้อยแล้ว ดังนั้นในโปรแกรมเตรียมตัวให้พร้อม (Set up) จะไม่มีชื่อเครื่องพิมพ์ EPSON LX 800 ในกรณีนี้คู่มือของเครื่องพิมพ์จะแนะนำกลุ่มของเครื่องพิมพ์ เช่น คุณอาจจะเลือกจาก LX 86, LX 80, RX 80, FX 80 หรือถ้าเป็นเครื่องพิมพ์ของ EPSON เช่นกันแต่เป็นรุ่นตระกูล LQ 850 (LQ 1050), LQ 500, LQ 2550, LQ 800 (LQ 1000) หรืออาจจะเลือกจากตัวอักษรที่เป็นรุ่นตามลำดับดังนี้ EX, FX, LX, RX, MX เป็นต้น

แต่สิ่งหนึ่งที่ยังไม่ได้กล่าวถึงเลยนั้นคือ เครื่องพิมพ์ประเภทเลเซอร์ ในที่นี้ผมขออิงกับเครื่องพิมพ์ของฮิวเล็ตต์-แพคการ์ด (HEWLETT PACKARD) รุ่นเลเซอร์เจ็ตซีรี่ส์ทู (LASER JET SERIES II) ตามคู่มือที่มากับเครื่อง ถ้าเราศึกษาดูจะพบว่า เป็นเครื่องพิมพ์ที่ออกแบบมารับคำสั่งเป็นคาแรคเตอร์ (character) แทนที่จะเป็นไบนารี่เหมือนกับเครื่องพิมพ์แบบเข็ม ตามคู่มือบอกว่าเป็นลักษณะของ Printer Command Language (PCL) ซึ่งทางบริษัทเป็นผู้พัฒนาขึ้นเป็นมาตรฐานทางภาษาสำหรับควบคุมการทำงานของเครื่องพิมพ์เลเซอร์ กล่าวคือ ไบต์แรกจะเริ่มด้วย 27 และปิดท้ายด้วยตัวอักษรตัวใหญ่เสมอ เช่น 27 & l1 o แต่ถ้าในบรรทัดต้นเป็นกลุ่มตัวเดียวกันสามารถเขียนรวมกันได้ เช่น 27 & l 1 o + 27 & l 2 A กลายเป็น 27 & l 1 O 2 A เป็นต้น จากที่ได้ทดลองใช้เครื่องพิมพ์เลเซอร์ในตอนแรกรู้สึกว่าจะยุ่งยากพอสมควร เนื่องจากว่าเครื่องพิมพ์แบบเข็มนั้นคำสั่งบางคำสั่งจะมี NULL คาแรคเตอร์ (แอสกีศูนย์) คำสั่งหยุดการตีเส้นใต้แต่ในแง่ของภาษาซีแล้ว จะมองว่าตัวไบนารี่ศูนย์นั้นคือจุดจบของชุดอักษร (STREAM OF CHARACTER) ทำให้การพิมพ์รายงานออกมาไม่สมบูรณ์ เนื่องจากโปรแกรมเข้าใจผิดคิดว่าจบชุดอักษรแล้ว แต่เราก็มีทางออกโดยการใช้คำสั่งอื่นแทน การพิมพ์ชุดตัวอักษรแอสกีศูนย์ประสมอยู่ สำหรับเครื่องพิมพ์แบบเลเซอร์นั้นจะไม่มีปัญหาเลย ขอยกตัวอย่างเกี่ยวกับกระดาษที่จะใช้กับเครื่องพิมพ์เลเซอร์ ปกติกระดาษที่จะใช้กับเครื่องพิมพ์แบบเลเซอร์นั้นมีวิธีการอยู่ 4 วิธี ขอยกตัวอย่างคำสั่งเลยนะครับ

27 & l # H

โดยที่ # = 0 (เลขศูนย์) ให้พิมพ์หน้ากระดาษปัจจุบัน
1 ป้อนกระดาษจากถาดอเนกประสงค์ (MP)
2 ป้อนกระดาษด้วยมือโดยใช้ถาดอเนกประสงค์
3 ป้อนซองจดหมายด้วยถาดอเนกประสงค์
4 ป้อนกระดาษจากอุปกรณ์เพิ่มเติมจากตลับบรรจุกระดาษ (OPTIONAL LOWER CASSETTE)

หากเราประสงค์ให้เครื่องพิมพ์เลเซอร์พิมพ์ข้อมูลที่หน้ากระดาษปัจจุบันที่ยังอยู่ภายในเครื่อง โดยการส่งคำสั่ง 27 & l 0 H ในแง่ของภาษาซี เราสามารถที่จะทำได้ง่ายขึ้น เช่น

```
Void PaperSource (int mode).
{   Char buffer [7];
    Sprintf (buffer, "%d&l%d", 27, mode);
}   fputs (buffer, stdprn);
```

จากตัวอย่างฟังก์ชัน PaperSource รับค่า mode มาแล้วจัดรูปแบบของคำสั่งไว้ในบัฟเฟอร์ก่อน แล้วก็ส่งออกพิมพ์ นับว่าเป็นจุดเด่นของ PCL ที่รับคำสั่งในรูปของแอสกีแทนการรับคำสั่งในรูปของไบนารี่ จุดเด่นต่อมาของเครื่องพิมพ์แบบเลเซอร์คือ มีรูปแบบฟอนต์ให้เราเลือกใช้มากมาย ทั้งนี้ขึ้นอยู่กับอุปกรณ์เพิ่มเติมคือหน่วยความจำที่ขยายเพิ่มเติม แล้วแต่เราจะเลือกฟอนต์จากคาร์ทริดจ์จากดิสเกตต์ เราสามารถที่จะใช้โหมดการทดสอบดูจะมีชื่อประจำของฟอนต์มากมาย แต่ถ้าเราเลือกใช้ฟอนต์จากดิสเกตต์จะเสียเวลาที่ใช้ในการดาวน์โหลดไปนานพอสมควร เครื่องพิมพ์เลเซอร์หากเรามีฟอนต์ตัวจิ๋วเราก็ส่งคำสั่งให้เครื่องพิมพ์นำฟอนต์ตัวจิ๋วมาใช้ สำหรับเครื่องพิมพ์แบบเข็มนั้นเราจะส่งเลขไบนารี่ 15 (หรือ 00001111) ไปให้เครื่องพิมพ์ แต่สำหรับเครื่องพิมพ์เลเซอร์ขึ้นอยู่กับตอนที่เราดาวน์โหลด หากว่าเรากำหนดหมายเลขฟอนต์ตัวจิ๋ว เช่น เป็นเลข 37 การที่จะใช้ฟอนต์ตัวจิ๋วเราก็จะต้องส่งคำสั่งขอเลือกใช้ฟอนต์ไปให้ ในที่นี้คือ 27 (37x รวมทั้งหมด 6 ไบต์ ดังนั้นคำถามที่ว่าเราจะนำเครื่องพิมพ์เลเซอร์มาใช้กับโปรแกรมสำเร็จรูปได้หรือไม่ สามารถตอบได้แน่นอนถึง 90% ว่า ถ้าโปรแกรมสำเร็จรูปไม่ระบุว่าใช้กับเครื่องพิมพ์เลเซอร์ได้ละก็จะไม่สามารถพิมพ์ได้ด้วยสาเหตุดังนี้

1. มีคำสั่งที่ใช้กับเครื่องพิมพ์ชนิดเข็มแตกต่างโดยสิ้นเชิง
2. หากข้อความเป็นภาษาไทยยิ่งตอบได้ว่า รายงานที่พิมพ์ออกมาจะไม่เหมือนกับที่ใช้เครื่องพิมพ์แบบเข็ม

ขอย้อนกลับมาดูโปรแกรมที่เราได้ลงทุนลงแรงงานสมอง และเวลาไปว่าได้เตรียมการอย่างไรที่จะแก้ปัญหานี้ หากว่าจะให้เรามาแก้ไขโปรแกรมอยู่ตลอดเวลาเท่ากับว่าเรายิ่งผูกให้โปรแกรมของเราติดกับอุปกรณ์ฮาร์ดแวร์มากขึ้น เรามาดูว่าทำอย่างไรจึงจะเป็นวิธีการที่ดีที่สุด และโปรแกรมของเราอ่อนตัวสามารถรับสถานการณ์ที่ทาง

ฮาร์ดแวร์จะพัฒนาไป คงจะมีอยู่หลายวิธีที่จะให้โปรแกรมของเราคอมแพตกับฮาร์ดแวร์ แนวทางที่เป็นไปได้ก็คือ

1. แบ่งโปรแกรมเป็นโมดูลและมีโมดูลเฉพาะเรื่องของการส่งข้อมูลออกทางเครื่องพิมพ์

2. ตกลงกันภายในทีมทำงานถึงมาตรฐานของคำสั่งที่จะสั่งเครื่องพิมพ์ หรือเราตั้งชื่อว่า คำสั่งเทียม

3. กำหนดหรือจัดกลุ่มรูปแบบของคำสั่งที่จะส่งให้กับเครื่องพิมพ์ เช่น แบ่งเป็น 5 กลุ่มดังที่ได้กล่าวมาแล้ว (จะไม่ขอกล่าวซ้ำ) เรามาดูรายละเอียดเพิ่มเติมนะครับ

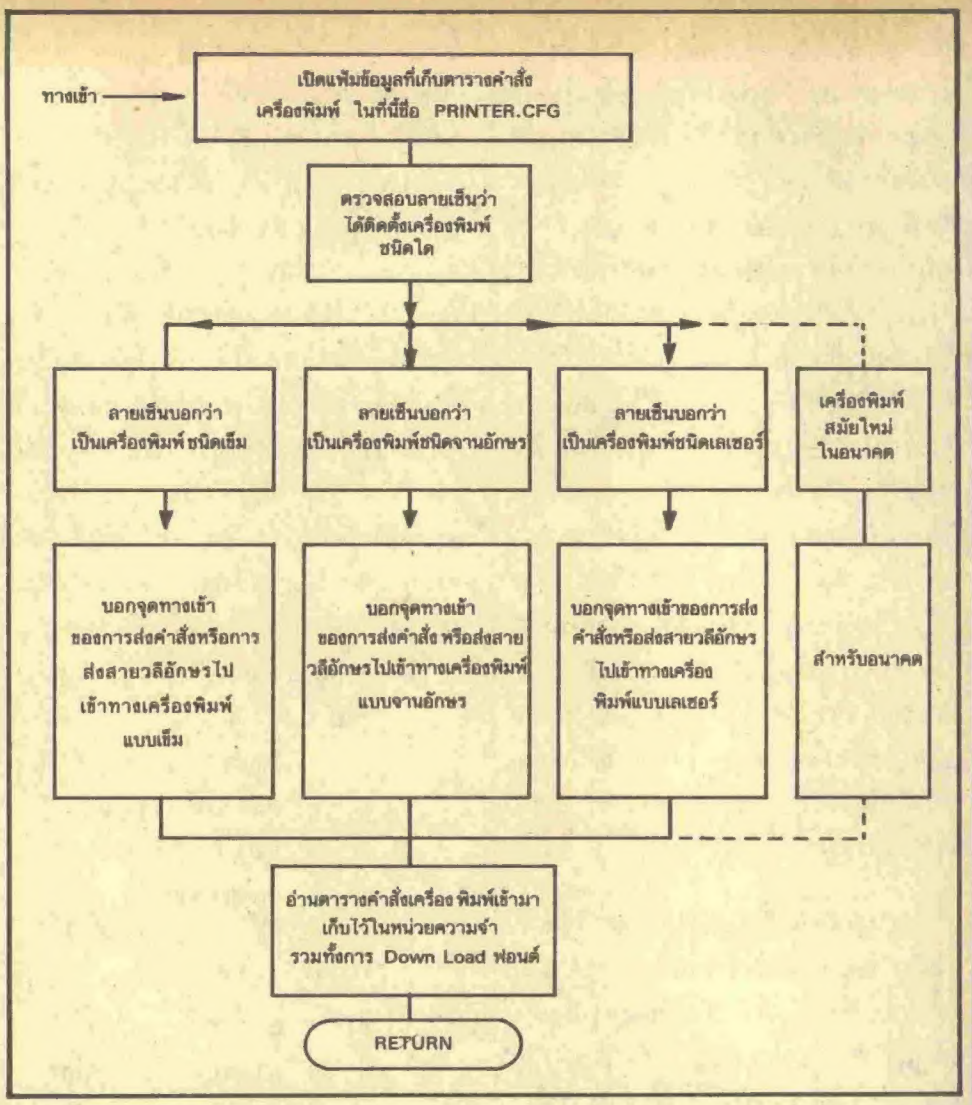
แบ่งโปรแกรมเป็นโมดูล

การแบ่งโปรแกรมเป็นโมดูล และมีโมดูลเฉพาะเรื่องของการส่งข้อมูลออกทางเครื่องพิมพ์ ความหมายในที่นี้เราพอจะวิเคราะห์ได้ว่า ควรจะมี 4 โมดูล กล่าวคือ

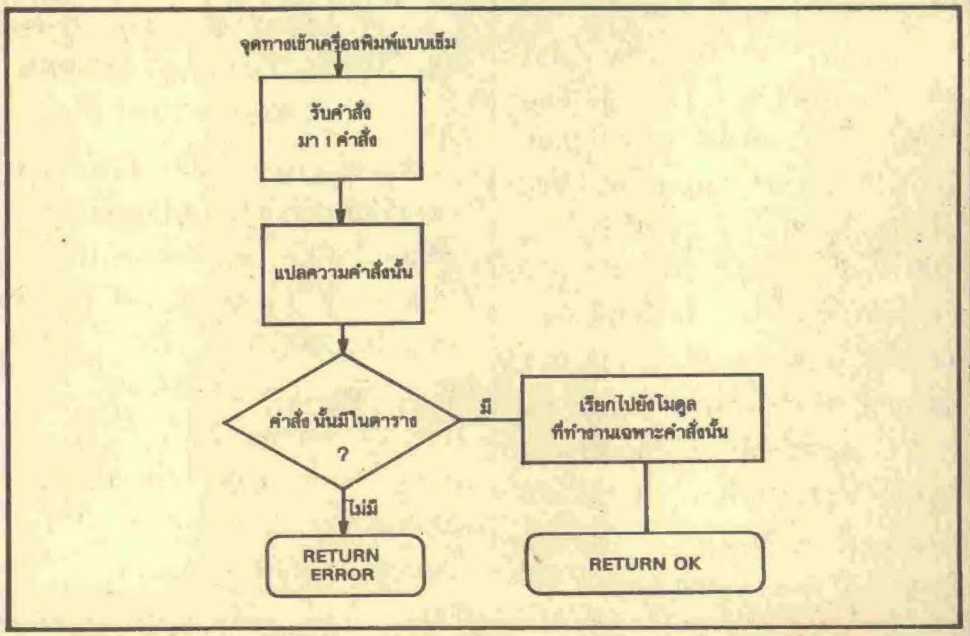
1. โมดูลการเริ่มต้น โมดูลนี้ทำหน้าที่คือ อ่านไฟล์ที่เก็บตารางคำสั่งเทียมและคำสั่งของเครื่องพิมพ์ พร้อมกับตรวจดูว่าตารางที่อ่านเข้ามาเก็บไว้นี้จะใช้กับเครื่องพิมพ์ประเภทเข็มหรือเลเซอร์ ปรับฟังก์ชันการทำงานให้ถูกต้อง ดังรูปที่ 1

2. โมดูลของคำสั่งที่ส่งให้กับเครื่องพิมพ์ หน้าที่ของโมดูลนี้ไม่ต้องเก็บคำสั่งของเครื่องพิมพ์ เมื่อรับคำสั่งเทียมมานำคำสั่งมาเทียบกับตาราง แล้วทะยอยนำค่าจากตารางส่งให้เครื่องพิมพ์ไป เช่น คำสั่งขึ้นกระดาษแผ่นใหม่เป็นต้น ดังรูปที่ 2

3. โมดูลส่งชุดตัวอักษรให้กับเครื่องพิมพ์ หน้าที่ของโมดูลนี้ทำหน้าที่อ่านตัวอักษรในชุดตัวอักษรครั้งละ 1 ตัว ตัวอย่างเช่น เวิร์ดสตาร์ (WORD STAR) จะใช้ดอตเป็นจุดเริ่มต้นของคำสั่ง ดังนั้น โมดูลนี้เมื่อตรวจพบดอตก็จะดูไบต์ต่อไปว่าเป็นคำสั่งหรือไม่ ถ้า



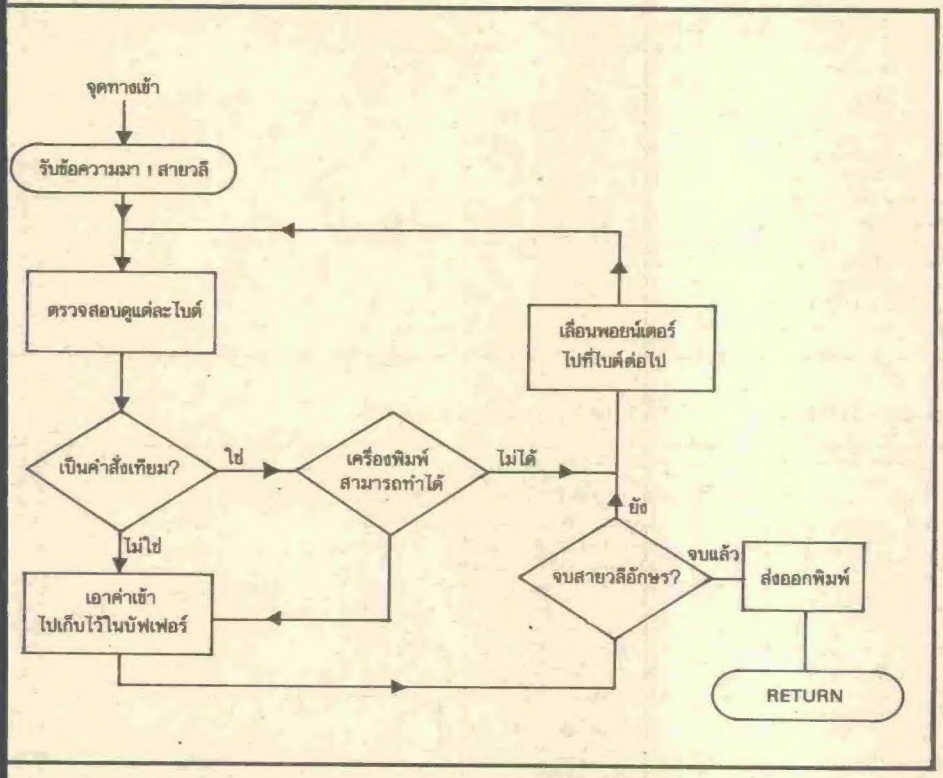
รูปที่ 1 โมดูลเริ่มต้น



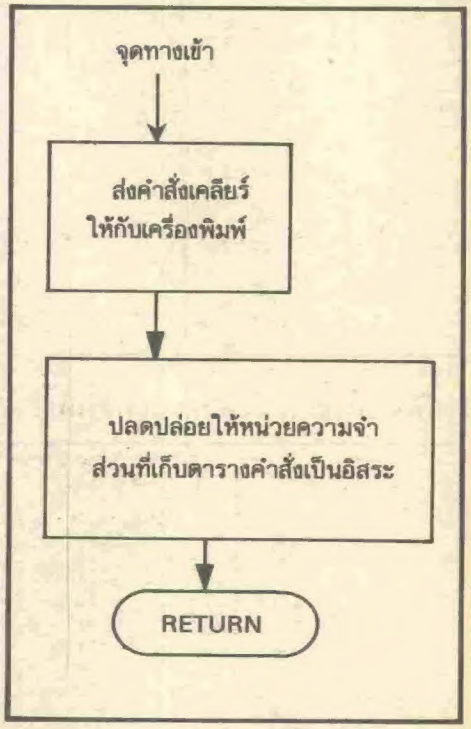
รูปที่ 2 โมดูลคำสั่งที่ส่งไปให้เครื่องพิมพ์ (ยกตัวอย่างเครื่องพิมพ์แบบเข็มส่วนเครื่องพิมพ์แบบอื่น เช่นเครื่องพิมพ์แบบเลเซอร์จะมีลักษณะแบบเดียวกัน)

ช่ก็นำไปเทียบกับค่าในตาราง แล้วนำค่าในตารางเข้ามาแทนที่ชุดตัวอักษรตำแหน่งคำสั่งเทียม แล้วจึงจะตรวจสอบไบต์ต่อ ๆ ไปจนกระทั่งจบชุดตัวอักษร นกรณีของตัวหนังสือภาษาไทยก็สามารถทำได้พร้อม ๆ กัน กล่าวคือจัดภาษาไทยไปด้วยหากพบคำสั่งเทียมก็เอาค่าจากตารางเข้ามาแทนที่คำสั่งเทียม

4. โมดูลยกเลิกการพิมพ์ โมดูลนี้มีหน้าที่สุดท้าย หมายความว่าเมื่อเราจะจบโปรแกรมควรจะเคลียร์ (CLEAR) เครื่องพิมพ์ให้กลับสู่สภาพเพาเวอร์ออน (POWER ON) เพื่อปล่อยให้หน่วยความจำที่เราได้อ่านค่าตารางคำสั่งเข้ามาเก็บไว้เป็นอิสระ (FREE MEMORY) ดังรูปที่ 4



ที่ 3 โมดูลส่งชุดตัวอักษรให้กับเครื่องพิมพ์ (ส่วนเครื่องพิมพ์ชนิดอื่นก็มีลักษณะเดียวกัน)



รูปที่ 4 โมดูลยกเลิกการพิมพ์ FREE MOMORY และ CLEAR PRINTER

ที่ 5 ตารางแสดงให้เห็นระหว่างคำสั่งเทียมของโปรแกรมและคำสั่งของเครื่องพิมพ์ชนิดเข็ม

```
/*------ PRINTER LX800 ---------*/
\?1LX800 CHR      ;Dowload font (Draft)
\A0               ;Dot matrix printer
\@  27 64         ;Clear printer
\D0 27 85  0      ;Cancel Unidirection
\D1 27 85  1      ;Select Unidirection
\B0               ;Select 0=KU or 1=SMO code
\F  27 12         ;Form feed
\Hxx 27 67 0 F11  ;Set page lenght in inch (default 11')
\J0xxx 27 74 F216 ;Perform precision 1/216 inch
\J1xxx 27 74 F47  ;Perform n/216 inch line #1 spacing
\J2xxx 27 74 F105 ;Perform n/216 inch line #2 spacing
\J3xxx 27 74 F100 ;Perform n/216 inch line #3 spacing
\J4xxx 27 74 F42  ;Perform n/216 inch line #4 spacing
\lxxxx 27 108     ;Set left margin (default=0)
\M1               ;Download printer draft mode
\O0 27 56         ;Disable paper out sensor
\O1 27 57         ;Enable paper out sensor
\P10 18 27 80     ;Select 10 pitch characters (default)
\P12 18 27 77     ;Select 12 pitch characters
\P17 15 27 80     ;Select 17 pitch characters
\P20 15 27 77     ;Select 20 pitch characters
\Q0 27 120  0     ;Select Draft mode (default)
\Q1 27 120  1     ;Select NLQ mode
```

ลายเซ็นต์ที่บอกว่าเราได้ติดตั้งเครื่องพิมพ์แบบเข็มให้กับระบบ

รหัสเครื่องพิมพ์แบบเกษตร

รูปที่ 5 (ต่อ)

```
\B0 27 116 0        ;Select to standard character set
\B1 27 116 1        ;Select to IBM graphics characters
\C0 18      แม็พตาราง;Cancel condense print
\C1 15              ;Select condense print
\D0 27 85 0         ;Cancel Unidirection
\D1 27 85 1         ;Select Unidirection
\f1 128 129 130 131 132 133 134 135 136 137 138 ;Define character font box
\G0 27 72           ;Cancel double strike
\G1 27 71           ;Select double strike
\I0 27 53           ;Cancel Italic character
\I1 27 52           ;Select Italic character
\K1 27 75           ;Select 8 bit Single density (60 dots/inch)
\K2 27 76           ;Select 8 bit Double density (120 dots/inch)
\K3 27 90           ;Select 8 bit Quadruple density (240 dots/inch)
\N0 27 37 0         ;Cancel down load character
\N1 27 37 1         ;Select down load character (Draft)
\R  27 58 0 0 0     ;Copy char font from ROM to RAM
\S0 27 83 1         ;Select Subscript
\S1 27 83 0         ;Select Super script
\S2 27 84           ;Cancel Super/Subscript
\T1 27 8            ;Back space one chr Next chr will over write
\T2 27 8 47         ;Back space one chr and over write With chr(47)
\U0 27 45 0         ;Cancel Under line (default)
\U1 27 45 1         ;Select Under line
\W0 27 87 0         ;Cancel Double-width print
\W1 27 87 1         ;Select Double-width print
```

รูปที่ 6 ตารางแสดงให้เห็นระหว่างคำสั่งเทียมของโปรแกรมและคำสั่ง ของเครื่องพิมพ์เลเซอร์

```
/*------ HPLaser.prn --------*/
\A2                 ;Printer laser  ลายเซ็นต์ที่บอกว่าเราได้ติดตั้งเครื่องพิมพ์ แบบเลเซอร์ให้กับระบบ
\?19THAIL050.SFP    ;Normal font 05 cpi     Resume Draf mode ID=19
\?10THAIP050.SFP    ;Normal font 10 cpi     Resume Draf mode ID=10
\?12THAIE042.SFP    ;Normal font 12 cpi     Resume Draf mode ID=12
\?17THAIC050.SFP    ;Normal font 16.67 cpi  Resume Draf mode ID=17
\?29THAIL150.SFP    ;Normal bold 05 cpi     Resume NLQ mode  ID=29
\?20THAIP150.SFP    ;Normal bold 10 cpi     Resume NLQ mode  ID=20
\?27THAIC150.SFP    ;Normal bold 16.67 cpi  Resume NLQ mode  ID=27
\?30THAIP250.SFP    ;Italic font 10 cpi     Resume Draf mode ID=30
\?57THAIC030.SFP    ;Scrip font 16.67 cpi   Resume Draf mode ID=57
\@   27 E           ;Clear printer
\R1                 ;Select SMO code        รหัสเครื่องพิมพ์เป็น สมอ.
\F   12             ;Form feed
\Hxx 27 &l%dA F2    ;Set page length in inch (default 8.5x11 )
\Y1  F3150          ;xx= 1 -> Excutive (7.25x10.5')
\Y2  F3300          ;  = 2 -> Latter (8.5x11')
\Y3  F4200          ;  = 3 -> Legal (8.5x14')
\Y26 F3507          ;  =26 -> A4 (210mmx297mm)
\Y80 F2250          ;  =80 -> Monarch Envelope(3 7/8 x 7 1/2')
\Y81 F2850          ;  =81 -> COM-10 Envelope(4 1/8 x 9 1/2')
\Y90 F2598          ;  =90 -> International DL (110mm x 220mm)
\Y91 F2704          ;  =91 -> International C5 (162mm x 229mm)
\J0  27 *p%dY F300  ;Perform 300 dpi
\J1xx F40           ;Perform 10/300 dpi
\J2xx F110          ;Perform 20/300 dpi
\J3xx F120          ;Perform 50/300 dpi
\J4xx F66           ;Perform 20/300 dpi
                    ;xxx==feed spacing in 1/1000 inch (in ASCII code)
\lxxxx 27 &a%dL F0  ;Set left margin (default)
                    ;xxxx==total space in 1/1000 inch (in ASCII code)
\P10 27 (%dX F10    ;Select 10 cpi              ID=10
\P12 F12            ;Select 12 pitch characters ID=12
\P17 F17            ;Select 17 pitch characters ID=17
\Q0  F10            ;Select Draft               ID=10
\Q1  F20            ;Select Bold                ID=20
;
\C0  F10            ;Cancel condense print      ID=10
\C1  F17            ;Select condense print      ID=17
\I0  F10  แม็พตาราง ;Cancel Italic character    ID=10
\I1  F30            ;Select Italic character    ID=30
\f1 152 153 154 155 149 150 156 158 159 157 151 ;Map box character
\S1  F57            ;Select script char         ID=57
\T1   8             ;Back space one character Next character will over write
\T2   8 47          ;Back space one character and over write With chr(47)
\U0  27 &d@         ;Cancel Under line (default)
\U1  27 &d0D        ;Select Under line
\W0  F10            ;Cancel Double-width print  ID=10
\W1  F19            ;Select Double-width print  ID=19
\B0  27 (8U 27 (%dX ;Select to Thai character set ID=10
\B1  27 (10U 27 (s0p10.00h12.00v0s0b3T ;Select to IBM graphics characters set
```

ทั้ง 4 ขั้นตอนของการแบ่งโปรแกรมเป็นโมดูลก็สามารถควบคุมงานการพิมพ์ไว้ทั้งหมดครบถ้วน หากคุณดูโฟว์ (FLOW) ทั้ง 4 รูปอีกครั้งคงช่วยให้เข้าใจดีขึ้น

คำสั่งเทียม

ตกลงกันภายในทีมทำงานถึงมาตรฐานของคำสั่งที่จะส่งให้เครื่องพิมพ์ หรือเราตั้งชื่อว่า คำสั่งเทียม โดยกำหนดตัวอักษรอะไรก็ได้เป็นที่ยอมรับกันภายในทีมทำงาน ในคณะทำงานที่ผมร่วมกันทำงานอยู่นั้นได้ตกลงกันว่าใช้ "\" เป็นคำสั่งล่วงหน้า ผมขอเรียกว่า โทเก้น (TOKEN) แล้วยังกำหนดต่อไปด้วยว่า หากต้องการจะพิมพ์เครื่องหมาย "\" จะต้องเพิ่มเข้ามาเป็น "\ \" จะได้ "\" เพียง 1 ตัวอักษร แบบเดียวกับภาษาซีเพื่อว่าโปรแกรมจะได้ทำการแยกได้ดูกว่าเป็นคำสั่งของเครื่องพิมพ์หรือไม่หรือว่าต้องการส่งข้อมูลตัว "\" ออกไปพิมพ์ ขอให้ดูในรูปที่ 5 สำหรับเครื่องพิมพ์แบบเข็ม และรูปที่ 6 สำหรับเครื่องพิมพ์แบบเลเซอร์ประกอบ เป็นตัวอย่างของการนิยามมาตรฐานของคำสั่งสำหรับโปรแกรมทำการสร้างข้อมูลส่งมาให้กับโมดูลการพิมพ์โดยที่ตารางในรูปที่ 5 เป็นตารางคำสั่งเทียมอยู่ทางซ้ายมือกับคำสั่งของเครื่องพิมพ์จริง ๆ อยู่ทางขวา ส่วนตารางในรูปที่ 6 ก็เช่นเดียวกัน หากคุณดูตารางในรูปที่ 5 และตารางในรูปที่ 6 ไล่ไปทีละคำสั่ง คุณจะพบว่าในระดับของโปรแกรมแล้วจะง่ายในการเขียนและการดูแลบำรุงต่อไปในอนาคต เพราะว่าทุก ๆ โปรแกรมที่ต้องการจะพิมพ์ข้อความหรือส่งคำสั่งให้กับเครื่องพิมพ์นั้นจะมีมาตรฐานของคำสั่งภายในกลุ่มนั้น คือจะใช้แต่โทเก้นและคำสั่ง 1-3 ไบต์ ส่วนคำสั่งของเครื่องพิมพ์จริง ๆ นั้นจะเป็นการเทียบตาราง (Look up Table) และเอาค่าที่ตรงกันในตารางส่งให้เครื่องพิมพ์

```
example 1. to draw box or table
any text 1   \f1┌──────┬──────┐\f0 continue to next line
any text 2   \f1│\f0 text 1 \f1│\f0text 2  \f1│\f0
any text 3   \f1├──────┼──────┤\f0
any text 3   \f1│\f0 text 3 \f1│\f0 text 4 \f1│\f0
any text 4   \f1└──────┴──────┘\f0

example 2. mix line box is converted
           \f1╔═══════╦═══════╗\f0
           \f1║\f0       \f1║\f0       \f1║\f0
           \f1╠═══════╬═══════╣\f0
           \f1║\f0       \f1║\f0       \f1║\f0
           \f1╚═══════╩═══════╝\f0

example 3. line box for IBM graphic charecter
           \B1┌──────┬──────┐\B0
           \B1│      │      │\B0
           \B1├──────┼──────┤\B0
           \B1│      │      │\B0
           \B1└──────┴──────┘\B0

print in Superscrip / subscript mode :
* DRAFT  \S1Superscript\S2 / \S0Subscript\S2
\U1* DRAFT & UNDERLINE \S1Superscript\S2 / \S0Subscript\S2\U0
\Q1* NLQ    \S1Superscript\S2 / \S0Subscript\S2\Q0
```

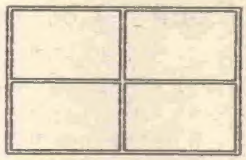
รูปที่ 7 ตัวอย่างข้อมูลที่ส่งให้กับโมดูล

example 1. to draw box or table

any text 1 ┌──────┬──────┐ continue to next line
any text 2 │ text 1 │ text 2 │
any text 3 ├──────┼──────┤
any text 3 │ text 3 │ text 4 │
any text 4 └──────┴──────┘

example 2. mix line box is converted

example 3. line box for IBM graphic charecter



print in Superscrip / subscript mode :

* DRAFT Superscript / Subscript

* DRAFT & UNDERLINE Superscript / Subscript

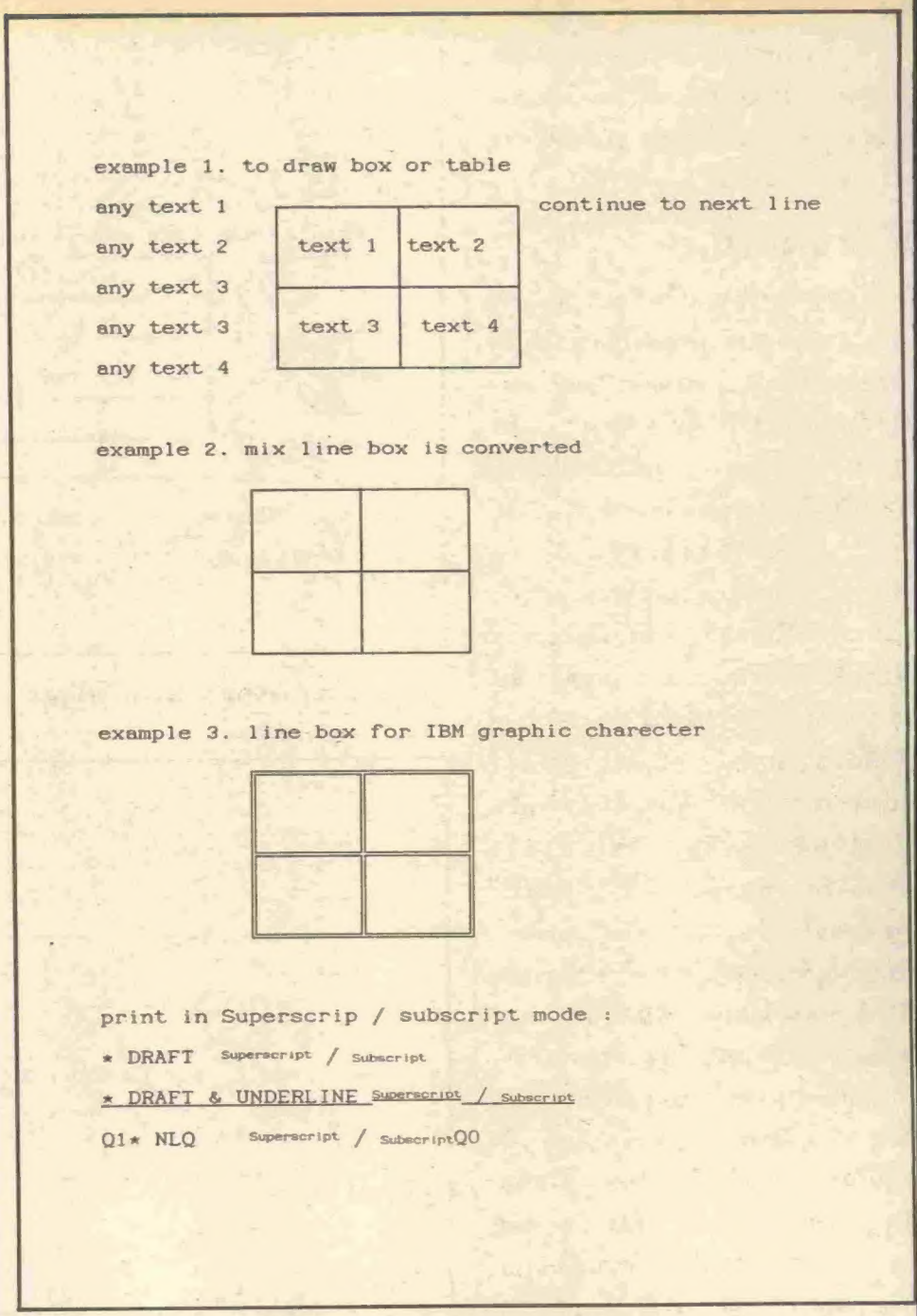
* NLQ Superscript / Subscript

รูปที่ 8 ตัวอย่างของข้อมูลที่พิมพ์ด้วยเครื่องพิมพ์แบบเข็ม LX 800

อีกครั้งในโมดูลการพิมพ์ ต่อไปในอนาคตหากว่ามีเครื่องพิมพ์รุ่นใหม่เข้ามา เราก็มีหน้าที่เพียงแต่ศึกษาว่าเข้ากับวิธีการดังกล่าวข้างต้นหรือไม่ หากว่าไม่สามารถเข้ากันได้ เราก็เพิ่มเติมส่วนของโปรแกรมโมดูลการพิมพ์เท่านั้นเอง ไม่ต้องมาไล่หาว่าการส่งพิมพ์นั้นโปรแกรมเมอร์เขียนไว้ที่บรรทัดใดในโปรแกรม

ในตารางในรูปที่ 5 และตารางในรูปที่ 6 เป็นตัวอย่างไฟล์ที่เก็บคำสั่งของเครื่องพิมพ์ชนิดเข็มกับคำสั่งของเครื่องพิมพ์เลเซอร์ เรามาศึกษาดูซัก 2–3 คำสั่งดู คำสั่งเทียม \A เราจะพบว่าในตารางในรูปที่ 5 ใช้ \A0 หมายถึงเครื่องพิมพ์แบบเข็ม แต่ในตารางในรูปที่ 6 ใช้ \A2 หมายถึงเครื่องพิมพ์แบบเลเซอร์ นอกจากนั้นเรายังสามารถที่จะบอกให้โปรแกรมรับรู้ได้ว่าเครื่องพิมพ์ที่เราใช้อยู่นั้นเป็นรหัสอะไร ในที่นี้ใช้ \E0 หมายถึงรหัสของเกษตร ส่วน \E1 หมายถึงรหัสของสมอ. ดังนั้นภายในคณะทำงานหรือทีมทำงานตกลงร่วมกันว่า ข้อมูลที่จะแก้นั้นใช้รหัสอะไร เช่น เราเก็บข้อมูลด้วยรหัสสมอ. แต่เครื่องพิมพ์เราใช้รหัสเกษตร ในโมดูลการพิมพ์จะต้องทำการแม็พรหัสหรือไม่ขึ้นอยู่กับคำสั่งเทียม \E อีกตัวอย่างหนึ่งคือ \f ผมออกแบบไว้สำหรับการทำตาราง คุณจะเห็นว่าสิ่งหนึ่งที่สามารถทำได้คือเอาค่าที่ตามมาจาก \f นั้นเป็นตัวตาราง ไม่ว่าผู้ขายเครื่องพิมพ์จะใช้รหัสเครื่องพิมพ์อย่างไร เราก็จะสามารถแม็พให้ตรงกับเครื่องพิมพ์ได้

รูปที่ 7 เป็นตัวอย่างของข้อมูลที่จะทำการส่งไปให้เครื่องพิมพ์ จะประกอบไปด้วยคำสั่งเทียม โดยเฉพาะการพิมพ์ตาราง รูปที่ 8 และ 9 แสดงให้เห็นว่า เครื่องพิมพ์ไม่ว่าจะเป็นของบริษัทไหนเราสามารถที่จะพิมพ์ออกรายงานได้ และประหยัดเวลาไม่ต้องมาคอยแก้ไขโปรแกรมตามเครื่องพิมพ์



รูปที่ 9 ตัวอย่างของข้อมูลเดียวกันแต่พิมพ์ด้วยเครื่องพิมพ์เลเซอร์

สรุป

การเขียนโมดูลสำหรับงานพิมพ์นั้นเป็นการแก้ปัญหาเครื่องพิมพ์ที่มาจากต่างบริษัท และรหัสไม่ตรงกับข้อมูลที่เราใช้เก็บลงดิสก์หรือแสดงที่หน้าจอ อาจจะเป็นการแก้ไขเราก็แก้ไขเฉพาะแฟ้มข้อมูลที่เก็บตารางคำสั่งเครื่องพิมพ์แทนที่จะคอยตามแก้ไขโปรแกรมซึ่งจะง่ายกว่า ส่วนข้อเสียก็มีคือทำให้โปรแกรมโตขึ้น เนื่องจากโปรแกรมยาวมากไม่สามารถนำมาลงพิมพ์ได้ หากคุณสนใจโปรแกรมทางวารสารได้บรรจุไว้ซอฟต์เมล์ฉบับนี้แล้วครับ